PHONE CALL

FOR TRACY OR LYNNE D. 11/03 TIME 8:50 A.M. P.M.
M ELEANOR COOK
OF US PATENT OFFICE ATTN: 571
PHONE/MOBILE (571) 272-9797 FAX ELEANOR COOK 273 0052
MESSAGE FAX SENT ON 10/31 5 PAGES DID NOT GO THROUGH -
DOCUMENT # 09173838
SIGNED JW

- [x] TELEPHONED
- [ ] RETURNED YOUR CALL
- [x] PLEASE CALL
- [ ] WILL CALL AGAIN
- [ ] CAME TO SEE YOU
- [ ] WANTS TO SEE YOU

OLFELD LLP
TY LAW
Street
019
)-712-0263

MARK HAYNES
ERNIE BEFFEL
WARREN WOLFELD
JIM HANN
Of Counsel:
*BILL KENNEDY
KENTA SUZUE
PETER SU
JOSEPH E. ROOT
*admitted in MA only

TTAL FORM

FAX NUMBER:	(571) 273-0052
TRANSMITTED TO:	Ms. Eleanor Cook
OF:	U.S. Patent Office, Board of Patent Appeals & Interferences
FROM:	Ernest J. Beffel, Jr.
CLIENT/MATTER:	Application No. 09/173,838 (OIN 1004-1)
DATE:	2 November 2006

TOTAL NUMBER OF PAGES INCLUDING THIS COVER SHEET:	12

ORIGINAL:	
x	WILL NOT FOLLOW
	FOLLOWS VIA U.S. MAIL
	FOLLOWS VIA COURIER

Dear Ms. Cook:

In connection with the above-identified U.S. patent application and pursuant to our telephone conversation of this morning, I am resending the pages that you did not receive in the original transmission sent to the Board of Patent Appeals & Interferences on 31 October 2006.

Kindly confirm that you have received the attached pages. We thank you for your assistance in this matter. Should you require anything else in this regard, please do not hesitate to contact me directly at (650) 712-0340, Ext. 204.

Best regards,

Lynne M. Milliot

Lynne M. Milliot
On behalf of Ernest J. Beffel, Jr.

* * * Memory TX Result Report (Nov. 2. 2006 9:41AM) * * *

1) FB DEHN
2)

e/Time: Nov. 2. 2006 9:38AM

File No.	Mode	Destination	Pg(s)	Result	Page Not Sent
5139	Memory TX	15712730052	P. 12	OK	

Reason for error
E. 1) Hang up or line fail
E. 2) Busy
E. 3) No answer
E. 4) No facsimile connection
E. 5) Exceeded max. E-mail size

HAYNES BEFFEL & WOLFELD LLP
INTELLECTUAL PROPERTY LAW
P.O. Box 366, 751 Kelly Street
Half Moon Bay, CA 94019
Phone 650-712-0340 Fax 650-712-0263
www.HMBlaw.com

MARK HAYNES
ERNEST BEFFEL
WARREN WOLFELD
JIM HANN
Of Counsel
*PHIL KRONROW
KENTA SUZUKI
PETER SO
JOSEPH H. RENTT
*admitted in MA only

FACSIMILE TRANSMITTAL FORM

FAX NUMBER:	(571) 273-0052
TRANSMITTED TO:	Ms. Eleanor Cook
OF:	U.S. Patent Office, Board of Patent Appeals & Interferences
FROM:	Ernest J. Beffel, Jr.
CLIENT/MATTER:	Application No. 09/173,838 (OIN 1004-1)
DATE:	2 November 2006

TOTAL NUMBER OF PAGES INCLUDING THIS COVER SHEET:	(12)

ORIGINAL:	
x	WILL NOT FOLLOW
	FOLLOWS VIA U.S. MAIL
	FOLLOWS VIA COURIER

Dear Ms. Cook:

In connection with the above-identified U.S. patent application and pursuant to our telephone conversation of this morning, I am resending the pages that you did not receive in the original transmission sent to the Board of Patent Appeals & Interferences on 31 October 2006.

Kindly confirm that you have received the attached pages. We thank you for your assistance in this matter. Should you require anything else in this regard, please do not hesitate to contact me directly at (650) 712-0340, Ext. 204.

Best regards,
Lynne M. Millot
On behalf of Ernest J. Beffel, Jr.

* * * Memory TX Result Report (Oct. 31. 2006 2:50PM) * * *

1) FB DEHN
2)

Date/Time: Oct. 31. 2006 2:34PM

File No.	Mode	Destination	Pg(s)	Result	Page Not Sent
5130	Memory TX	15712730299	P. 58	OK	

Reason for error
E. 1) Hang up or line fail
E. 2) Busy
E. 3) No answer
E. 4) No facsimile connection
E. 5) Exceeded max. E-mail size

HAYNES BEFFEL & WOLFELD LLP
INTELLECTUAL PROPERTY LAW
P.O. Box 366, 751 Kelly Street
Half Moon Bay, CA 94019
Phone 650-712-0340 Fax 650-712-0263
www.HMBay.com

MARK HAYNES
ERNIE BEFFEL
WARREN WOLFELD
JIM HANN
Of Counsel
*BILL KINNIGUY
KIRNTA SUZUKI
PETER SU
JOSEPH E. ROUT
*Admitted in MA only

FACSIMILE TRANSMITTAL FORM

FAX NUMBER:	(571) 273-0299
TRANSMITTED TO:	Board of Patent Appeals and Interferences
OF:	U.S. Patent and Trademark Office
FROM:	Ernest J. Beffel, Jr.
CLIENT/MATTER:	Application No. 09/173,858 (OIN 1004-1)
DATE:	31 October 2006
TOTAL NUMBER OF PAGES INCLUDING THIS COVER SHEET:	58

ORIGINAL:	
x	WILL NOT FOLLOW
	FOLLOWS VIA U.S. MAIL
	FOLLOWS VIA COURIER

CERTIFICATE OF FACSIMILE TRANSMISSION
I hereby certify that this correspondence is being facsimile transmitted to the U.S. Patent & Trademark Office, Board of Patent Appeals and Interferences, at Facsimile No. 571-273-0299 on 31 October 2006.

Lynne M. Milliot

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of: Bart Alan MELTZER et al.	Group Art Unit: 2178
Application No. 09/173,858	Examiner: HUYNH, Cong Lac T.
Confirmation No. 4734	
Filed: 16 October 1998	CUSTOMER NO. 22470
Title: **Documents for Commerce in Trading Partner Networks and Interface Definitions Based on the Documents**	

MAIL STOP APPEAL BRIEF - PATENTS
COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL REQUEST FOR RECONSIDERATION

Sir:

In connection with the above-referenced U.S. patent application, attached herewith are the following documents for filing:

[X] Appeal Request for Rehearing (with attachments); and
[X] Petition Under Rule 183 for Oral Argument on Rehearing.

The Commissioner is hereby authorized to charge any fees in relation to this document to our Deposit Account No. 50-0869 (Attorney Docket No. OIN 1004-1).

Respectfully submitted,

Dated: 31 October 2006

Ernest J. Beffel, Jr., Reg. No. 43,489

HAYNES BEFFEL & WOLFELD LLP
P.O. Box 366
Half Moon Bay, CA 94019
Telephone: (650) 712-0340
Facsimile: (650) 712-0263

CERTIFICATE OF FACSIMILE TRANSMISSION
I hereby certify that this correspondence is being facsimile transmitted to the U.S. Patent & Trademark Office, Board of Patent Appeals and Interferences, at Facsimile No. 571-273-0299 on 31 October 2006.

Lynne M. Milliot

Lynne M. Milliot

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:

Bart Alan MELTZER et al.

Application No. 09/173,858

Confirmation No. 4734

Filed: 16 October 1998

Title: **Documents for Commerce in Trading Partner Networks and Interface Definitions Based on the Documents**

Group Art Unit: 2178

Examiner: HUYNH, Cong Lac T.

CUSTOMER NO. 22470

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL REQUEST FOR REHEARING

Sir:

This Appeal Request for Rehearing under Rule 41.52 is filed in response to the unfavorable decision of August 31, 2006. Rehearing is appropriate because (1) the Board did not affirm the Examiner on any grounds specified by the Examiner regarding the Rule 131 declarations and raised new issues that should be briefed; and (2) the Board has overlooked certain elements of the claims that are missing from the 645-word McKendrick reference, which is a popular press report of potential uses for XML

that does not provide an enabling disclosure or written description of the claim elements, even in view of W3C's specification for XML.

Rehearing this case could put it on track for identifying the allowable subject matter, advancing a very old case. Progress in the prosecution has occurred haltingly and always with the support of those acknowledged to have special expertise in the issues presented. For instance, after multiple office actions and at least one RCE, the Section 101 issue in the case was resolved in a five-minute interview with Examiner (now Director) Jack Harvey, who was on the Section 101 panel. Applicants hope that the Board considers rehearing in light of appellants' repeated efforts to work with this Examiner, as apparent from the IFW.

As there has not been any oral hearing of this matter, this request is accompanied by a petition for oral hearing, so that further misapprehensions of the record on appeal and of the art of record can be avoided. This case is particularly prone to misapprehensions because the briefings are difficult to understand.

Should it be determined that any fees are required with regard to the filing of this document, the Commissioner is hereby authorized to charge those fees to Deposit Account No. 50-0869.

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Cases

Rules

Other Authorities

I. PARTICULAR STATEMENT OF GROUNDS FOR REHEARING

Rehearing, with oral argument, should be granted on two issues:

(1) Whether the Board's Rule 131 declarations prove reduction to practice of all elements in both independent claims 1 and 61 and require removal of McKendrick as a reference against claims 1 and 61?

(2) Whether the Board erred when it failed to discuss the representative claims as a whole and took a Section 103 position that, even if accepted, does not read on claims 1 and 61?

Rehearing is appropriate on the Rule 131 declarations because the Board's decision does not affirm the Examiner on any grounds specified by the Examiner before or during appeal. *See* Rule 41.50(a)(1) (regarding affirmance on grounds specified by the Examiner). The Board's decision should be treated under the residual Rule 41.50(b), even though the Board did not acknowledge that it was offering new grounds of rejection, because it presents a different rationale and raises different factual questions than the Examiner specified either before or during appeal. Rehearing is a matter of fundamental fairness.

Oral argument would be particularly helpful regarding the Rule 131 declarations, because they need to be read from the perspective of one with skill in the Web services art and familiarity with the general evolution of Web services technology. Counsel has worked on applications filed by the now defunct Commerce One for many years and can readily answer the Board's questions.

Rehearing is also appropriate on application of the McKendrick reference, in view of W3C, because the Board overlooked[1] that claim 1 is a data structure claim that specifies a particular interface definition (which, incidentally, has been widely adopted for Web services) and presented a rationale that does not read on all elements of claim 1 or 61. The Board's decision overlooks the difference between using XML documents and defining the claimed interface for use of XML documents.

Oral argument would also be helpful regarding application of the McKendrick and W3C references both because of the technology and the Examiner's confused

[1] We use "misapprehended" and "overlooked" throughout this request because the rules require this characterization. No offense is intended.

arguments. For lack of oral argument, significant questions regarding technological evolution from object-oriented architectures to XML documents to Web services went unanswered.

II. ARGUMENT

For the Board's convenience and to minimize the need for reference to the appeal briefs, we reproduce with emphasis the representative claims and excerpts from the declarations. The claims read:

> *1. An interface for transactions among nodes in a network including a plurality of nodes which execute processes involved in the transactions, the interface being stored in a computer readable medium, comprising:*
>
> *a machine readable specification of* ***an interface to transaction processes stored in memory*** *accessible by at least one node in the network,* ***including interpretation information providing a definition of an input document, and a definition of an output document****, the definitions of the input and output documents comprising respective descriptions of sets of storage units and logical structures for the sets of storage units.*
>
> *61. A method for programming a commercial transaction in a network, comprising:*
>
> *defining* ***a machine readable definition of an input document for a node*** *in the network including resources to execute a process in the transaction, and* ***a machine readable definition of an output document for the node****, the definitions of the input and output documents comprising respective descriptions of sets of storage units and logical structures for the sets of storage units; and*
>
> ***providing interpretation information*** *for the logical structures to the node.*

Claim 1 is a data structure device and claim 61 is a method.

The inventors declared that:

> *3. Prior to March 11, 1998,* ***we had implemented a registry*** *for trading partners.* ***The registry was used in a method****, also implemented prior to March 11, 1998, in a form* ***sufficient to demonstrate that the method would work for its intended purpose****, for establishing transactions among trading partners in a network, comprising: maintaining a registry of machine-readable specifications specifying business services offered by trading partners, the* ***machine-readable specifications including*** *at least one of definitions of, and references to definitions of, services offered and at least one of definitions of, and references to* ***definitions of, documents to be exchanged*** *with such services by trading partners; and* ***providing, in response to a request, one or more of the***

machine-readable specifications from said registry is via a communication network to a requesting node.

4. Attached hereto as Exhibit A is an excerpt of a memorandum, which I am informed, or know from personal knowledge, was written by co-inventor Glushko, prior to March 11, 1998. Exhibit A includes the statement, ***"In particular, the eCo server has now subsumed the registry and query services that had been envisioned as part of the Taxonomy of Everything in our proposal." This comment establishes that the registry and supporting services had been implemented*** *at the time the memorandum was written.*

To be clear about evidentiary status of paragraph 4, co-inventor Glushko was one of the declarants. An excerpt from Exhibit A reads:

> CNgroup has made substantial progress in both during the first quarter.
>
> CBL (Common Business Language) enables semantic interoperation and integration of different commerce applications. CBL defines the metadata for making a business and its services a self-describing "eCo component"; it enables the intelligent query and aggregation of product catalogs and descriptions; it represents the forms and messages needed for commercial transactions; and it can be used to "wrap" formats and messages to make legacy applications "eCo-compliant". Specific technical activities performed during the first quarter as part of CBL R&D included:
>
> 1. Development of a "design philosophy" for overall scope and approach of CBL
> 2. Analysis of existing standards for common information types and semantic primitives. Where appropriate, semantics have been drawn from the UN/EDIFACT Basic Semantic Unit data dictionary and certain ISO and IETF standards (e.g., for geographical location, date and time, currency, weights and measures).
> 3. Analysis of proposed metadata frameworks for Internet resources (Dublin Core, RDF, MCF).
> 4. Analysis of semantics of commerce as embodied in EDI X12 transaction sets, Uniform Commercial Code, and in proposals like the Open Buying on the Internet specification.
> 5. Creation of first draft of CBL to support the requirements of Project Seital (described above in "Project Baseline").
> 6. Determining an approach for CBL support of industry applications (and for ATP demonstrations in particular).
>
> The development of CBL has strongly shaped the requirements for the eCo runtime platform. XML is now at the core of the eCo architecture, and the eCo server can be thought of as an XML processing platform on which CBL is the reference application. The use of XML inside the eCo platform as well as in its applications has enabled the server to be more capable and extensible than we conceived at the time of the proposal.
>
> In particular, the eCo server has now subsumed the registry and query services that had been envisioned as part of the Taxonomy of Everything in our proposal. The TOE was proposed as a scalable, distributed registry service for implementing Internet-based directory and translation services, and a key architectural building block and core task in the Phase 1 plan.

A. McKendrick is not available as a reference against representative claims 1 and 61. Therefore, reversal is appropriate as to all claims.

Appellants' position is that the Rule 131 declarations, by the Board's own analysis, should remove McKendrick as a reference regarding representative claims 1 and 61. The penultimate analysis (at 9) tacitly acknowledges that the declarations read on claims 1 and 61, by omitting the representative claims from the list of claims considered to be unsupported. In perspective, McKendrick published the reference in September 1998, which predates this application by only two to six weeks, depending on when it actually was published. A 113-page application with 16 sheets of figures and a pair of related applications filed the same day takes longer than that to prepare.

The Board's decision raises four issues regarding the declarations that need to be addressed, the most prominent being the Board's note (at 8-9) that the "first draft of CBL" was not submitted with the declarations. If the Examiner had ever raised this factual issue, a supplemental declaration could readily have been submitted under Rule 131 or 132. At this stage of proceedings, we instead turn to what one of ordinary skill in the art would know about "the first draft of CBL," as revealed by Google.

1. Appellants' response to the new factual issue raised in the Board's decision at 8-9, regarding "first draft of CBL."

The significance of the "first draft of CBL" issue (at 8-9) is that properly understanding the relationship of CBL to Exhibit A leads, in the following section, to the conclusion that Exhibit A reads on all elements of claims 1 and 61, including an interface definition data structure that specifies input and output documents.

Following the Board's lead, we used Google[2] to find a presentation by inventor Glushko that predates McKendrick, explains use of CBL as claimed, and is consistent with use of CBL in the status report, Exhibit A. We also have located scholarly descriptions of these inventors' work on CBL.

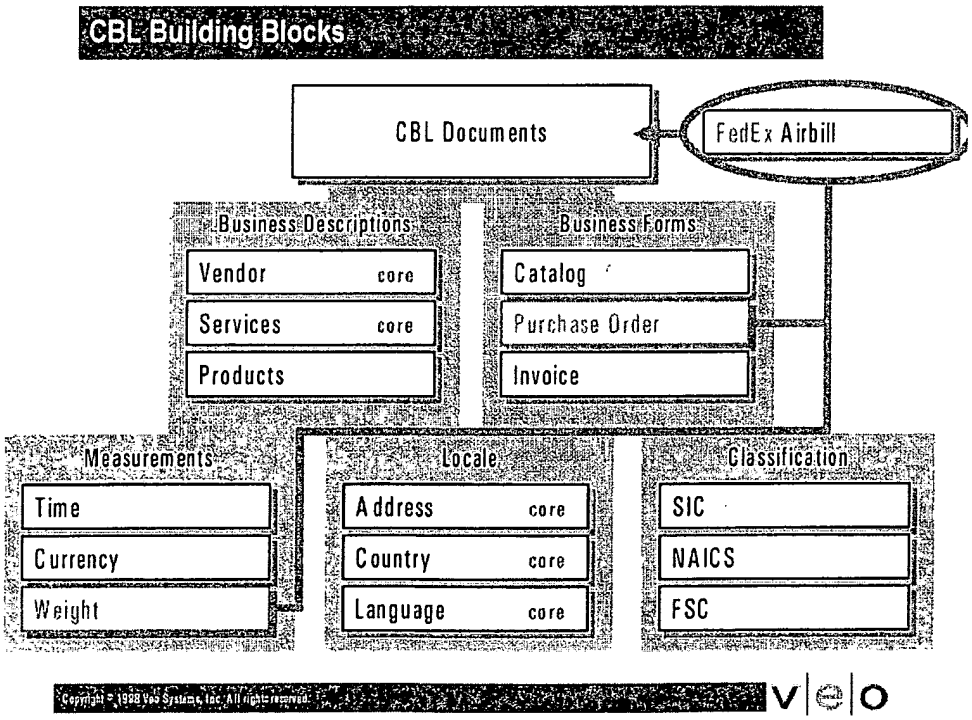
On July 25, 1998, the International Workshop on Component-based Electronic Commerce was held at the Fisher Center for Management & Information Technology,

[2] Commerce One declared bankruptcy years ago and liquidated its physical assets. The company's design and programming records were not transferred to the present assignee of record, Open Invention Network ("OIN"). The mission of OIN to preserve the Linux eco-system against potential IP challenges is generally described at **www.openinventionnetwork.com**.

Haas School of Business, UC Berkeley. The conference program, accessed at **http://groups.haas.berkeley.edu/citm/conferences/cec/** on October 26, 2006, lists inventor Meltzer as the keynote speaker and lists inventor Glushko as a speaker.

Glushko's slides refer to CBL document schemas in an interface definition data structure that includes input and output documents. Glushko, *Implementing Domain-specific Commerce Languages with a Common Business Library*, slides 29-31 (delivered July 25, 1998) accessed at **http://groups.haas.berkeley.edu/citm/conferences/cec/Presentations/Session3/glushko.pdf** on October 26, 2006. We reproduce and explain three of the slides, attaching all of the slides to this request for rehearing.

Purchase orders and invoices are both mentioned in "CBL Building Blocks" slide 29, which illustrates how Federal Express might use CBL to create an XML version of its airbill by customizing a generic purchase order DTD with specific information about shipping weight.



This slide matches FIG. 10 of the application.

Slide 30, "Business Services Described Using CBL", is an example of an interface definition data structure written in XML. The two service interfaces defined are named Submit Order and Track Order. The Submit Order service references an input "po.dtd" and an output "poack.dtd". These "dtd" files are document schemas. (Notice that this Submit Order service accepts a purchase order as input and returns an acknowledgement as output, not an invoice.)


```
<service>
<service.name>Order Service</service.name>
<service.location>www.veosystems.com/order</service.location>
<service.op>
        <service.op.name>Submit Order</service.op.name>
        <service.op.inputdoc>po.dtd</service.op.inputdoc>
        <service.op.outputdoc>poack.dtd</service.op.outputdoc>
</service.op>
< service.op>
        < service.op.name>Track Order</service.op.name>
        <service.op.inputdoc>request.track.dtd<service.op.inputdoc>
        <service.op.outputdoc>response.track.dtd<service.op.outputdoc>
</service.op>
</service>
```

This example ties perfectly to the disclosure in this application, as it matches LISTING 5 of the CD-ROM appendix to the revised (reformatted) specification, which appeared on page 45 of the original specification (before we moved source code to an appendix.)

Glushko's slide 30 responds, in part, to the Board's note about "first draft of CBL." It documents, two months before McKendrick, how the inventors were using CBL to define an interface and gives an example of an interface definition data structure for an order service that receives and acknowledges purchase orders.

Slide 31, "CBL status" explains that CBL was being used in demonstration applications. This corroborates the declarants' testimony in paragraph 3 that documents and registries were working for their intended purpose.

These three slides inform the Board as to how one of skill in the art would understand "CBL" in the status report exhibit, responsive to the Board's factual note. From Glushko's slides, one of ordinary skill in the art would recognize in the status report that these inventors were using CBL v1.0 as claimed, before McKendrick's popular press report.

Our search "glushko cbl 1998" also brought to our attention the date stamp 05/02/1998 at the end of LISTING 7 of the CD-ROM appendix, which appeared on page 86 of the original specification. All of the inventors submitted the original specification under oath, so this date stamp is testimony of record that code and a makefile predate McKendrick by several months. The date stamped code excerpt, at original specification page 82, appears to generate an XML service operation definition, including name, input and output. "Input" and "output" appear in the code.

Three additional references carry the date for the inventors' work on CBL back to 1997. A reference volume, Kenneth B. Sall, *XML Family of Specifications: A Practical Guide*, at 1073 (Addison Wesley 2002) provides the following glossary-like description of the xCBL e-commerce specification:

> Commerce One's XML Common Business Library promotes "cross-industry exchange of business documents such as product descriptions, purchase orders, invoices, and shipping schedules." Another goal is "to make the business documents, forms and messages that flow between businesses comprehensible to each business no matter what computer system is used." **The insightful CBL effort predates the XML 1.0 Recommendation, dating back to 1997.** xCBL uses a mature schema specification (SOX) which is a forerunner of the W3C's XML Schema Standard. See *http://www.xcbl.org/* for details.

(emphasis added).

A 1999 article by inventors Glushko and Meltzer and Dr. Jay Tenenbaum, with whom they worked, also gives 1997 as the year. Glushko, et al., *An XML Framework for Agent Based E-Commerce*, Communications of the ACM, Vol. 42, No. 3, pp. 106-114 (Mar. 1999). The authors explained their work, at 108: "Conceived originally as a CORBA-based interoperability framework, the eCo System architecture was **recast in 1997 on an XML foundation**, due to XML's simplicity and widespread adoption ..." This article, which we attach, explains Glushko slides 29 and 30, at pages 112-13.

In a textbook that he co-authored, Professor Glushko again referred to the 1997 work on CBL. "The earliest effort to attack the problem of semantic overlap among XML vocabularies for business applications was the XML **Common Business Library**, whose **first version was released in 1997**." Glushko and McGrath, *Document Engineering: Analyzing and Designing Documents for Business Informatics and Web Services*, at 130 (MIT Press 2005).

Having gone down this path and established a 1997 release of CBL v1.0, we need to point out that the passage to which the Board refers actually says "first draft of CBL to support the requirements of Project Seitai". We believe that Project Seitai involved Nippon Telephone and Telegraph (NTT) procurement. Slide 30 explains that CBL had been used to implement Project Seitai, a demonstration project, by July 1998, which is consistent with CBL v1.0 having been released in 1997.

With this information responsive to its newly raised factual note, how should the Board proceed? The Board has more than adequate proof to remove McKendrick as a reference and reverse the Examiner. Given the relative expertise of the Board and the Examiner in Rule 131 issues, reversal is preferred. Alternatively, it could remand the case to the Examiner with instructions to further consider the evidence in light of how it

would be understood by one of skill in the art and to accept additional evidence from appellants.

2. The Rule 131 declarations evince that a registry had been demonstrated to work for its intended purpose.

The Board opines (at 8) that appellants failed to provide a factual showing that the embodiment relied upon actually worked for its intended purpose. This part of the decision does not mention any elements of the claims. While the Board recites careful consideration of the evidence (at 8), its words (at 7-8) overlook the factual showing in paragraph 3 that the registry worked and the corroborating proof in paragraph 4 that the registry and supporting services had been implemented.

These inventors explicitly testified that they "had implemented a registry ... [that] was used in a method ... in a form sufficient to demonstrate that the method would **work for its intended purpose**." One of ordinary skill in the art, with an understanding of how these inventors were using CBL, would understand that these inventors implemented a registry of machine-readable specifications including documents to be exchanged as an interface definition data structure including input and output documents, before the critical date of McKendrick's article. We ask the Board to reconsider the probative weight of the testimony in paragraph 3.

In paragraph 4, quoted by the Board (at 6), the inventors declared, "Exhibit A includes the statement, 'In particular, the eCo server has now subsumed the registry and query services that had been envisioned as part of the Taxonomy of Everything in our proposal.' This comment establishes that the registry and supporting services had been implemented at the time the memorandum was written." To one of skill in the art, this is not a vague or general statement, because it focuses on a particular sentence in the accompanying status report. This concise explanation of the corroborating document reinforces the explicit testimony in paragraph 3. One of skill in the art, familiar with the publicly acclaimed efforts of eCo, Veo/Commerce One and W3C, would acknowledge paragraphs 3 and 4 as connecting the declarations, exhibit and claim language and proving that the embodiment relied upon actually worked for its intended purpose. We ask the Board to consider the testimony in paragraph 4.

Appellants urge that the Board has overlooked oral and documentary evidence of record or misapprehended how it would be understood by one of skill in the art. The

Examiner's refusal to give any weight to this evidence should be reversed, because the evidence proves that the embodiment relied upon would work for its intended purpose.

3. Exhibit A would be understood by one of skill in the art to include input and output documents.

The Board did not believe, when it wrote its decision (at 8), that Exhibit A reads completely on the language of the claims. The Board focuses on input and output documents. Our presentation of how those of skill in the art would understand the status report and reference to "CBL" informs our reading of Exhibit A and shows that input and output documents are referenced in the declarations.

One of skill in the art, looking back on the status report as an historical document, would be aware of CBL and the development efforts of eCo, Veo/Commerce One and W3C. With this awareness and understanding, it is plain that the status report Exhibit A, as explained in declarations paragraph 4, refers to a registry that included interface definition data structures having input and output document schemas.

Particularly, CBL is extensively discussed in the status report. Beta testing was at hand, in application of CBL to Project Seitai. The registry and query services had been implemented in the eCo server. One of skill in the art would recognize that the registry and supporting services included definitions of input and output documents, for instance, as illustrated in Glushko slide 30.

Appellants urge that the Board overlooked significant documentary evidence of record or misapprehended how it would be understood by one of skill in the art, which reads on the claims.

4. The Board misapprehended the law when it faulted appellants for not briefing how the declarations read on dependent claims 2-16 and 62-72.

In this section, we respond to the Board's penultimate rationale (at 9) that criticizes appellants for not reading the declarations on the dependent claims. This part of the decision tacitly acknowledges that all elements of the representative claims are covered. It does not identify any particular element of any dependent claim as not being covered. Appellants' position is that they had no burden on appeal to address limitations of any claims beyond the representative independent claims. Appellants' briefs properly focused on the issues argued by the Examiner before and during appeal and on the representative claims. There are two compelling reasons why it was not

necessary for appellants to anticipate in their briefs the Board's newly-raised dependent claims burden of proof issue.

First, the Examiner did not make the Board's burden of proof argument. What the Examiner wrote regarding the substance of the declarations (*Examiner's Answer* [corrected], at 18-19 (mailed Sept. 22, 2005)) was:

> Exhibit A, submitted as a written description, does not constitute an actual reduction to practice. Furthermore, only the filing of a US patent application which complies with the disclosure requirement of 35 USC 112 constitutes a constructive reduction to practice. A written description, no matter how complete, which has not been made the subject of a US patent application, does not qualify as reduction to practice. Accordingly, Applicants have not established prior invention. The rejection is maintained [sic].

This argument combined with the argument on page 17, "Applicants attorney can not argue that the evidence provided in the Exhibit supports the claimed limitations [because] [t]he evidence and facts must be either stated in the declaration or incorporated by reference thereto", amounts to a refusal by the Examiner to give any probative weight to either the testimony or corroborating exhibit. Appellants addressed the Examiner's issues and were not required to anticipate issues that were not raised below or anywhere in the Examiner's Answer.

As a matter of law, based on the *Ex parte Ovshinsky*, 10 U.S.P.Q.2d (BNA) 1075 (Bd. Pat. App. & Inter. 1989) case cited by the Board, the Examiner's erroneous refusal to give any weight to sworn testimony should require reversal. The Board cites *Ex parte Ovshinsky* as precedent for requiring a declaration to address every element of a claim. In that case, the Board reversed the examiner's refusal to consider the testimony in the declarations, which showed more than the accompanying exhibits. "This failure to give probative weight to the Rule 131 declarations constitutes reversible error. ... [I]t is entirely appropriate for appellant to rely on a showing of facts set forth in the Rule 131 declarations themselves to establish conception of the invention prior to the effective date of the reference. This appellants have done." There is no requirement in the case to address claims that are not selected for appeal.

The *Ex parte Ovshinsky* decision cites *Ex parte Swaney and Banes*, 89 U.S.P.Q. (BNA) 618 (Bd. Pat. App. & Inter. 1950), another case in which the Board reversed the

examiner and held that the declarations offered were sufficient, even without written corroboration of all the elements of the claims.

The facts in *Ex parte Ovshinsky* are on all fours with this case, because that examiner made the same mistake as this one: Both refused to give any evidentiary weight to the testimony in the declarations or the exhibit. The Board reversed that examiner and should reverse this one as well.

Second, the appellate rules call for designation of claims to be argued separately and for appellants to stick to the specified grouping of claims when writing their briefs. As both appellants and the Examiner argued two groups of claims, corresponding to the independent claims 1 and 61, the rules relieved appellants of any obligation to discuss the dependent claims. The decisions cited did not look at minor features of dependent claims that were not separately argued on appeal. The *Ex parte Swaney and Banes* decision discusses only the representative claim 20, just as we have discussed only the representative claims 1 and 61. The cases do not support the Board's form of analysis or argument. The rules make it clear that we should select representative claims and argue just those claims. Therefore, appellants should not be faulted for applying the declarations to representative claims 1 and 61, without discussing the dependent claims.

Accordingly, it was enough to have shown that the declarations are effective to remove McKendrick as a reference against the representative, independent clams 1 and 61. As a matter of law, appellants are entitled at least to a ruling that the reference is unavailable against claims 1 and 61 and to a remand for consideration of the dependent claims. The Board should remove the reference against the representative, independent claims, following through on what it tacitly acknowledged.

B. Because McKendrick is not available as a reference against representative claims 1 and 61, allowability of the independent claims makes the dependent claims allowable.

Every principle of patent law entitles applicants to a reversal of the Examiner's rejection of the dependent claims, once McKendrick has been removed as a reference against the independent claims, making the independent claims allowable over the art of record.

It is black letter law that, if the independent claims are valid over a reference, then the dependent claims are as well. "If an independent claim is nonobvious under 35 U.S.C. 103, then any claim depending therefrom is nonobvious. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988)." MPEP § 2143.03 at 2100-131 (8th Ed. Rev. 5 Aug. 2006).

The appellate rules, as interpreted by the Federal Circuit, dictate that narrower, dependent claims shall prevail on appeal if the broader, independent claims do. *In re Fritch*, 972 F.2d 1260, 1266 n. 17 (Fed. Cir. 1992) (reversing obviousness determination). The Federal Circuit opined, "when argued together, dependent claims stand or fall with the independent claims from which they depend." *Id.* This principle has been reiterated for many years. *In re Sernaker*, 702 F.2d 989, 991, 217 U.S.P.Q. (BNA) 1 (Fed. Cir. 1983); *In re Burckel*, 592 F.2d 1175, 1178-79, 201 U.S.P.Q. (BNA) 67, 70 (CCPA 1979).

We know of no rule or legal principle that required us to address dependent claims separately on appeal, to support analysis of the independent claims. To the contrary, the rules tell us to save the Board time and trouble by grouping dependent claims with independent ones. The Examiner did not assert any grounds of rejection (e.g., § 112) that required us to separately address or cancel the dependent claims and made no argument anywhere in the Examiner's answer regarding application of the declarations to the dependent claims.

The Board erred when it faulted appellants for not walking through the dependent claim elements in argument and erred again when it assumed[3] that elements the dependent claims are not covered by the declarations.

We urge the Board to follow the law and reverse rejection of the dependent claims, because the declarations and accompanying exhibit effectively remove McKendrick as a reference against the independent claims 1 and 61.

---

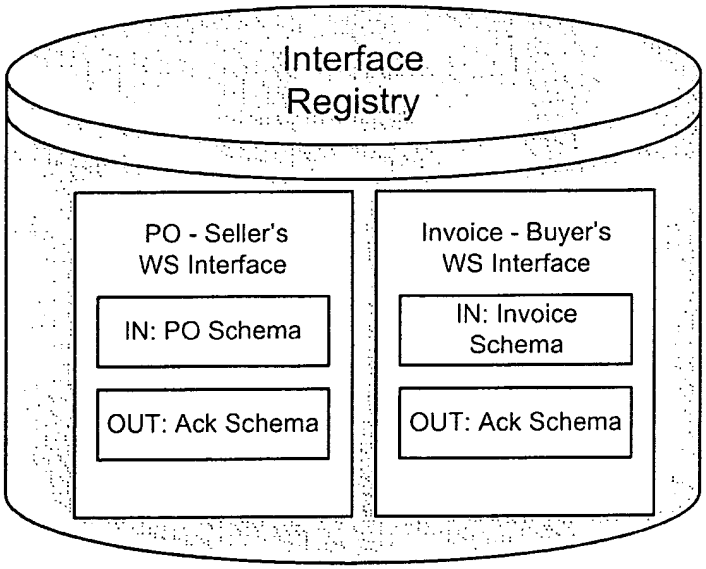
[3] *Decision*, at 9. This is clearly erroneous because, for instance, dependent claim 15 makes the same connection between XML and descriptions of sets of storage units and logical structures for the sets of storage units that the Board's decision makes. *Id.* at 12. Dependent claim 4 refers to a registry, which is covered in both the declarations and exhibit. And we could go on ...

C. When it applied section 103(a) to McKendrick in view of W3C, the Board misapprehended the claim as a whole, overlooked the interface definition data structure, and penned a rationale that fails to read on claim 1.

It is likely that the Board misapprehended claims 1 and 61, as its decision makes no mention of a data structure that defines an interface, referring instead to "interface definitions based on the documents". *Decision*, at 1. Appellants' position is that simply using XML documents does not read on an interface definition data structure that pairs input and output XML documents as a process interface.

1. Claimed technology illustrated in a Web services environment.

To illustrate the claimed technology, we have sketched an example including purchase order and invoice interfaces, responsive to the Board's reasoning. *Decision*, at 13. This sketch, set in an XML Web services environment, illustrates two instances of machine readable interface definition data structures. The first transaction process interface defines a seller's Web service that receives and acknowledges purchase orders. The second transaction interface defines a buyer's process that receives invoices and acknowledges receipt. Focusing on the seller's PO Web service interface, the interface definition data structure includes definitions of an input PO and an output acknowledgement document, corresponding to Glushko slide 30. The definitions are labeled schemas, because schemas are one way to define documents comprising respective descriptions of storage units and logical structures for the sets of storage units. Many alternative schema languages are available. A structure including an interface registry or repository is claimed in dependent claim 4. A method of programming using such an interface definition data structure appears in independent claim 61, with reference to providing a registry in dependent claim 64.

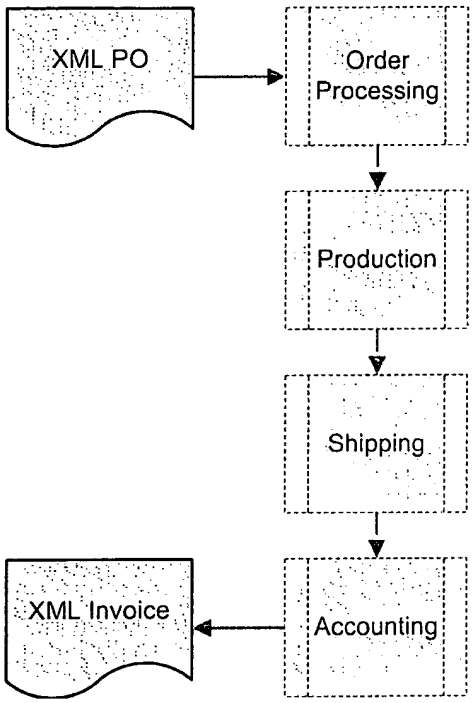


Because the Board's words do not portray interface definition data structures or an interface registry (*compare Decision*, at 1), we believe that the Board

misapprehended the claim as a whole and overlooked the interface definition data structure in the claims.

2. McKendrick reference illustrated from the Board's decision.

The McKendrick reference in view of W3C does not enable or provide a written description of an interface definition data structure or an interface registry, either as claimed or illustrated. Another sketch may help, this one drawn from the Board's characterization (at 13-14) of McKendrick's teachings. We illustrate in the left column a purchase order and an invoice, which are mentioned in McKendrick. *Decision* at 14. The business that receives the purchase order, produces and ships the product, then bills the customer is represented in the right column. Typical data processing subsystems for order processing, production, shipping (fulfillment) and accounting (billing) are depicted. The subsystems are indicated with dotted lines, because the McKendrick reference does not mention any of these subsystems. We do not illustrate an interface definition data structure because McKendrick does not mention any interface to which purchase orders are submitted or from which invoices are generated or any data structure.



3. Neither McKendrick nor W3C teaches a Web services interface.

As background, the Board should understand that Web service architectures are a pioneering technology because they loosely couple modules with well-defined, provider-agnostic interfaces. This is a paradigm shift from prior technologies such as CORBA, which were very tightly coupled, involved exchanging binary formatted objects instead of passable documents, and required negotiation of proprietary object structures. This paradigm shift is explained in the attached article, Glushko 1999.

One way of implementing Web services (a term that is not used in either the McKendrick or W3C reference) is to build the claimed interface definition data structure

and expose it in a public registry. Conversely, Web services are not the only way to use XML. For instance, XML can be used in e-mail message attachments.

Consider a more likely, pre-invention implementation of purchase orders and invoices using XML: In business, many things happen, such as production and shipping, and weeks pass between a PO and an invoice. Traditional process interfaces are short lived – system operators will not tolerate a transaction process that receives a PO as input and remains active for weeks until it can output a corresponding invoice. Traditional processes and interfaces are separated into distinct order processing and invoicing subsystems. Therefore, it would have been more likely, pre-invention, that an e-mail with an XML PO document attached would be sent from buyer to seller, followed weeks later by an e-mailed XML invoice document generated by a different process and different subsystem than the one that received the PO. Even post-invention, one of ordinary skill in the art would not understand an invoice to be an output of a PO receiving process.

It follows from the alternative e-mail implementation that the claimed interface definition data structure cannot be inherent in sending and receiving XML documents. *Ex parte Levy*, 17 U.S.P.Q.2d (BNA) 1461, 1464 (Bd. Pat. App. & Inter. 1990); *see, generally*, MPEP § 1212 at 2100-47 to 48. Therefore, a Web service-style interface with an interface definition data structure is not disclosed explicitly or inherently in either McKendrick, W3C or the combination.

4. The Board's reasoning does not read on all elements of claim 1.

The Board's decision (at 14) reasons through the application of McKendrick. The element missing from the rationale is an interface definition data structure. There is nothing inherent in the combination of McKendrick and W3C that produces the claimed data structure.

The gap in the Board's reasoning is apparent in its discussion of hindsight (at 15-16), which does not distinguish between using XML documents and an interface definition data structure that pairs input and output XML documents as a process interface.

An interface definition data structure specifying input and output documents, for instance Glushko slide 30, is not inherent in using XML documents, because there are alternative ways to use XML documents, as discussed above. The Board did not justify

(at 14) creating the claimed interface definition data structure out of whole cloth. The interface definition data structure is not taught, suggested or implied by either of the references, so it is not in the combination either, unless one applies hindsight.

To review the Board's reasoning (at 13-14), we constructed the following chart:

An interface for transactions among nodes in a network including a plurality of nodes which execute processes involved in the transactions, the interface being stored in a computer readable medium, comprising:	"the Internet includes 'a plurality of nodes' and ... web pages are stored on a computer readable medium (at 13:19-20)
A machine readable specification	"XML is a machine readable specification" (at 14:12-13)
providing a definition **of an interface to transaction processes stored in memory**, accessible by at least one node in the network, including	
providing a definition of an input document, and a definition of an output document,	"the instant claimed input and output documents broadly read upon McKendrick's XML purchase orders and invoices" (at 13-14) "McKendrick's XML purchase orders and invoices are clearly associated with corresponding XML declarations" (at 14:16-17)
the definitions of the input and output documents comprising respective descriptions of	XML DTD schema provides descriptions, per reply brief.
sets of storage units and logical structures for the sets of storage units.	This is XML, per reply brief.

What the Board says (at 14), which comes closest to discussing an interface definition data structure is, "McKendrick's use of XML (as defined by the W3C XML specification) to perform financial transactions on the Internet clearly meets the language of the claim that recites 'a machine readable specification of an interface to transaction processes stored in memory accessible to at least one node in the network, including interpretation information providing a definition of an input document, and a definition of an output document.'" To test this conclusory statement, we studied the rationale (at 13-14) and found that there is no logical support for jumping from a pair of

XML documents, even if they relate to the same transaction, to the claimed interface definition data structure. As explained above, a logic combination of McKendrick and W3C is to send POs and invoices via e-mail from separate processes. Therefore, the rationale does not rend an interface definition data structure as claimed.

Alternatively, POs and invoices are not input and output documents for a process, such as a Web service. One of skill in the art would, of course, read the application, be aware of Glushko slide 30 and Glushko 1999, and consider the meaning of a transaction interface with that awareness. All three references identify a purchase order acknowledgement as the output document from a transaction process that receives a PO as input. This is a practical necessity, if the PO receiving process is to have both an input and output document (Web services can be defined to have just an input) because of the potential latency between the PO and invoice. This is required for operational reasons described above and because businesses need to know that their orders have been received. Accordingly, we made the invoice as an input to the buyer's system, not as an output from the seller's system.

Therefore, the Board should reconsider and conclude that its rationale does not read on all elements of claim 1, because the combination does not include an interface definition data structure and because McKendrick's reference to POs and invoices would not be understood as input and output documents to the same process. As explained above, if claim 1 is allowable, so are claims 2-16

D. Rejection of claim 61 was improper because McKendrick does not read on the claim

Claim 61 is a programming method that adds to claim 1 the step of providing interpretation information for the logical structures to the node.

Preliminarily, appellants and the Board agree that the Examiner's rejection of claim 61 begins by admitting that McKendrick does not read on the claim, because the final office action at 14-15, began, "McKindrick [sic] ***does not disclose*** explicitly ... defining a machine readable definition of an input document for a node in the network including resources to execute a process in the transaction, and a machine readable definition of an output document for the node, the definitions the input and output documents comprising respective descriptions of sets of storage units and logical structures for the sets of storage units".

Appellants believe that the Board overlooked (at 15-16 and 18) alternative ways in which McKendrick might motivate an artisan to implement purchase orders and invoices on the Internet using XML. The Examiner admits that McKendrick teaches nothing about our way of using interface definition data structures. Nothing in W3C suggests how to put multiple documents together into an interface. Without evidence or rationale suggesting this particular way of combining the references, the Board is slipping into using the claim as a blueprint aided by 20-20 hindsight, which is impermissible. 2-5 Chisum on Patents § 5.03 [2][c] n. 29 (2005 Lexis version); *e.g. ATD Corp. v. Lydall, Inc.*, 159 F.3d 534, 546, 48 USPQ2d 1321, 1329 (Fed. Cir. 1998) ("Determination of obviousness cannot be based on the hindsight combination of components selectively culled from the prior art to fit the parameters of the patented invention."); *Grain Processing Corp. v. American Maize-Products Corp.*, 840 F.2d 902, 907, 5 USPQ2d 1788, 1792 (Fed. Cir. 1988) ("Care must be taken to avoid hindsight reconstruction by using 'the patent in suit as a guide through the maze of prior art references, combining the right references in the right way so as to achieve the result of the claims in suit.' ") Even a "compelling suggestion" (at 18) to implement purchase orders and invoices on the Internet using XML does not teach or suggest combining the references to achieve the claimed result, which includes an interface definition data structure that is not found in either reference.

For all of the reasons given above in the context of claim 1 and for the additional reason that neither the Examiner or the Board find the additional element of claim 61 in either of the references, we urge the Board to reconsider rejection of claim 61. Upon reconsideration of claim 61, the Examiner also should be reversed as to claims 62-72.

V. CONCLUSION

In view of the foregoing, Appellants ask that this honorable Board reconsider its decision and reverse the Examiner's rejections of the claims. In addition, it is submitted that all claims that are the subject of this examination are now allowable, and a notice of intent to issue a patent is respectfully requested.

Fee Authorization. The Commissioner is hereby authorized to charge any fee determined to be due in connection with this communication, or credit any overpayment, to our Deposit Account No. 50-0869 (File No. OIN 1004-1).

Respectfully submitted,

Dated: 31 October 2006

Ernest J. Beffel, Jr., Reg. No. 43,489
Attorney for Patent Owner

HAYNES BEFFEL & WOLFELD LLP
P.O. Box 366
751 Kelly Street
Half Moon Bay, CA 94019
Telephone: 650.712.0340
Facsimile: 650.712.0263

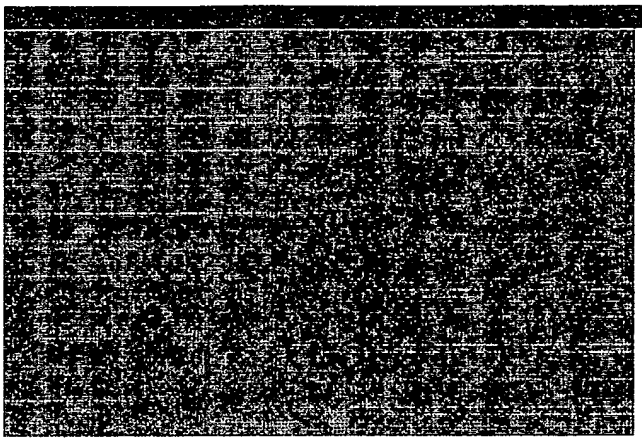
ATTACHMENT:

GLUSHKO 1998 PRESENTATION

v|e|o

Implementing Domain-specific Commerce Languages with a Common Business Library

Dr. Robert Glushko -- Director, Information Engineering



4005 Miranda Avenue, Suite 150
Palo Alto, CA 94304
415.858.7711 main
415.858.4925 fax
www.veosystems.com
info@veosystems.com

1

Outline of the Talk

- XML as a technology platform for commerce applications
- Domain-specific commerce languages
- A common business library

About Veo Systems

- **History**
  - 1/97 for-profit "spin-off" of CommerceNet Consortium called "CNgroup"
  - 9/97 received multi-million $ award from U.S. Commerce Department ATP to help commercialize "eCo" component-based commerce framework (along with CommerceNet)
  - 4/98 changed name from CNgroup -Veo
- **Status**
  - Privately held, backed by corporate and VC investors, growing very fast
  - Products later this year

veo

XML as Technology Platform

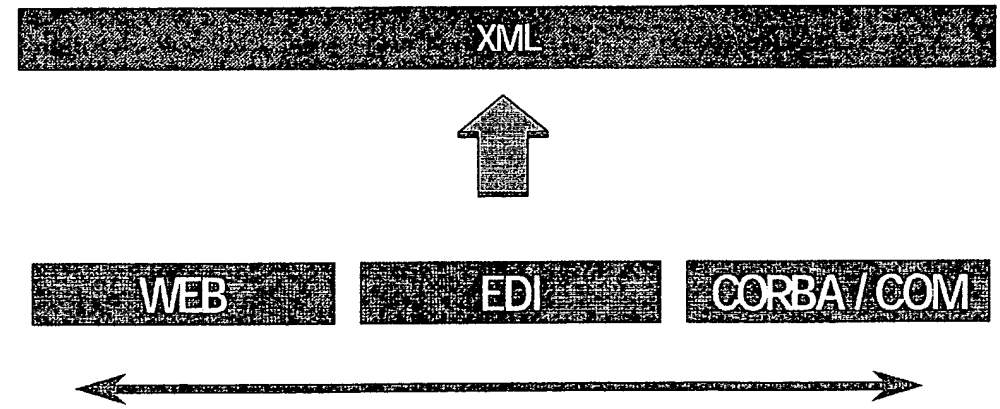
The XML Revolution

- Today's Web sites publish information for people
  - "eyeballs-only" is dominant design perspective
  - hard to search
  - hard to automate processing
    (too much "scraping and hoping")
- Tomorrow's sites will provide information and services for computers (and people)
  - overcomes HTML's inherent limitations
  - enables the new business models of the network economy

XML as Technology Platform

...exchange data in an application and vendor neutral format

...the simplicity of HTML with the precision of APIs



Commerce Networks ⟷ Shared Information Models

- Supply Chains
  - Merchants, distributors, manufacturers, brokers, logistics, shippers
- Real Estate
  - Brokers, banks, escrow, title, inspection, MLS, government agencies, classifieds, loan aggregators
- Securities
  - Brokers, financial advisors, markets, research services, account management
- Travel
  - Hotels, airlines, rental car agencies, travel agents

Laptop Description Seen "By Eye"

Laptop Computer

IBM Thinkpad 560X

233 Mhz

32 Mb

4 Gb

4.1 pounds

$3200

HTML Laptop Description

```
<TITLE>Laptop Computer</TITLE>
<BODY>
<UL>
<LI>IBM Thinkpad 560X
<LI>233 Mhz
 <LI>32 Mb
 <LI>4 Gb
 <LI>4.1 pounds
 <LI>$3200
</UL></BODY>
```


XML Laptop Description

```
<COMPUTER TYPE="LAPTOP">
<MANUFACTURER>IBM</MANUFACTURER>
<LINE>Thinkpad</LINE>
<MODEL>560X</MODEL>
<SPEED UNIT="MHZ">233</SPEED>
 <MEMORY UNIT="MB">32</MEMORY>
 <DISK UNIT="GB">4</DISK>
 <WEIGHT UNIT="POUND">4.1 </WEIGHT>
 <PRICE CURRENCY="USD">3200</PRICE>
</COMPUTER>
```

Smarter Processing Enabled by XML

- Shared schema for laptops, desktops, and towers
- <COMPUTER> provides a logical container for extracted and manipulating product information as a unit
  - Sort by <MANUFACTURER>, <SPEED>, <WEIGHT>, <PRICE>
- Explicit identification of each part enables its automated processing
  - Convert <PRICE> from "USD" to French Francs, Italian Lira, etc.

Airline Schedule Seen "By Eye"

Airline Schedule

Flight Information

United Airlines #200

San Francisco

11:30

Honolulu

2:30

$368.50

HTML Airline Schedule

```
<Title>Airline Schedule</Title>
<Body>
<H2>Flight Information</H2>
<H3>United Airlines #200</H3>
<UL><LI>San Francisco
<LI>11:30
<LI>Honolulu
<LI>2:30
<LI>$368.50
</UL></Body>
```


Airline Schedule in XML

```
<TransportSchedule  Type="Airline">
<Segment Id="United Airlines #200">
 <Origin>San Francisco</Origin>
<DepartTime TZ="PST">11:30 </DepartTime>
 <Destination>Honolulu</Destination>
<ArriveTime TZ="HST"> 2:30 </ArriveTime>
<Price Currency="USD">368.50</Price>
</Segment>
</TransportSchedule>
```

Example: Schema for Transport

Using the same schema for all scheduled transportation services:

<TransportSchedule Type="Airline">
<TransportSchedule Type="Train">
<TransportSchedule Type="Ferry">

An application could create itineraries that involve more than one service by matching on locations and times

Shared Semantics for Time and Location

Shared semantics for location and time in all schemas that need them enables richer "commerce networks" of services:

<TransportSchedule Type="Airline"> ...
<Destination>Honolulu</Destination>

<Accommodation Type="Hotel">...
<Destination>Honolulu</Destination>

<Event Type="Concert">...
<Destination>Honolulu</Destination>

v|e|o

Domain-Specific Commerce Languages

17

Domain-Specific Languages for Commerce Networks

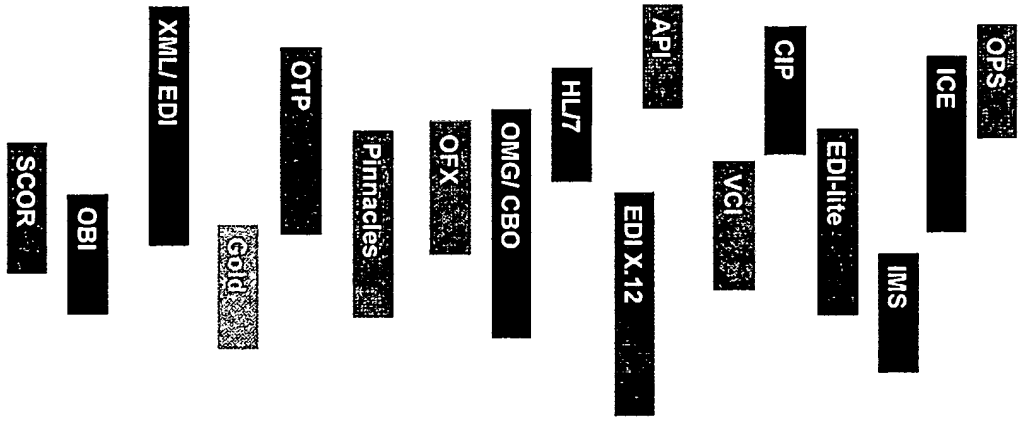
OBI	Corporate Procurement	AMEX, Office Depot, Boise Cascade
OTP	Retail Payment	Mastercard, Mondex
OFX / GOLD	Personal Finance	(Intuit, Microsoft), (IBM, 125 Banks)
ECOM	Computer Supply Chain	Ingram + 24 largest channel players
ICE	Content syndication	News Corp., Sun, Microsoft, Adobe, Vignette, C/Net

This list is growing explosively, and all are using XML (or shortly will be)...

XML and Metcalfe's Law

- XML makes it easy to create markup languages
- But the value of a language depends on how many people (or computers) understand it
- How do you encourage and enable others to understand your language?
- The EDI approach:
  - BIG COMPANY: Speak MY language or I won't do business with you!
  - SMALL COMPANY: Yes, master.
- The XML approach:
  - Excuse me, here are the rules of my language if you'd like to speak with me...

Tower of Babel - Stovepipe Protocols



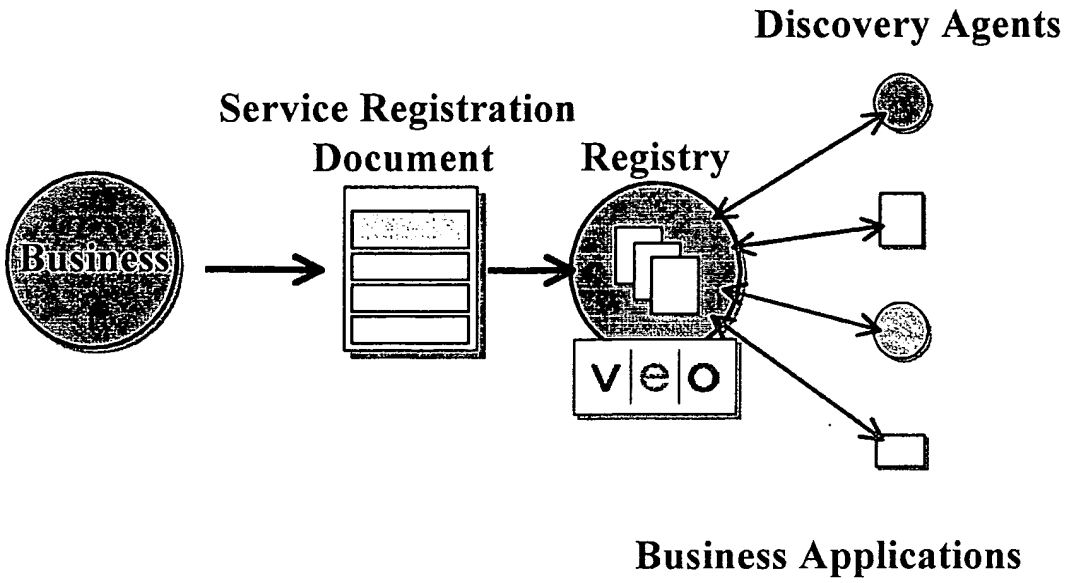
- Delayed time to market
- Redundant development costs
- Limited Interoperability

The Common Business Library

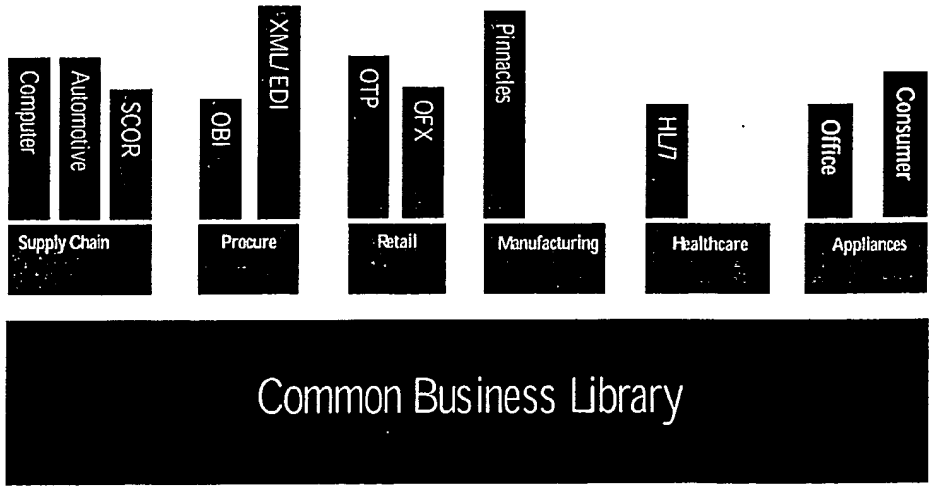
"Loose Coupling" via Shared Document Definitions

- Interconnect business systems and services in terms of the documents they exchange rather than in terms of their application interfaces
- Shared document definitions provide an intuitive framework for specifying the business logic and computations that take place on each end of the exchange.
- Five shared document definitions are implied in these two business rules:
  - if you send me a **request** for a catalog, I will send you a **catalog**
  - if you send me a **purchase order** and I can fulfill it, I will send you a **shipping notice** and an **invoice**

Businesses Publish Services Using XML Documents



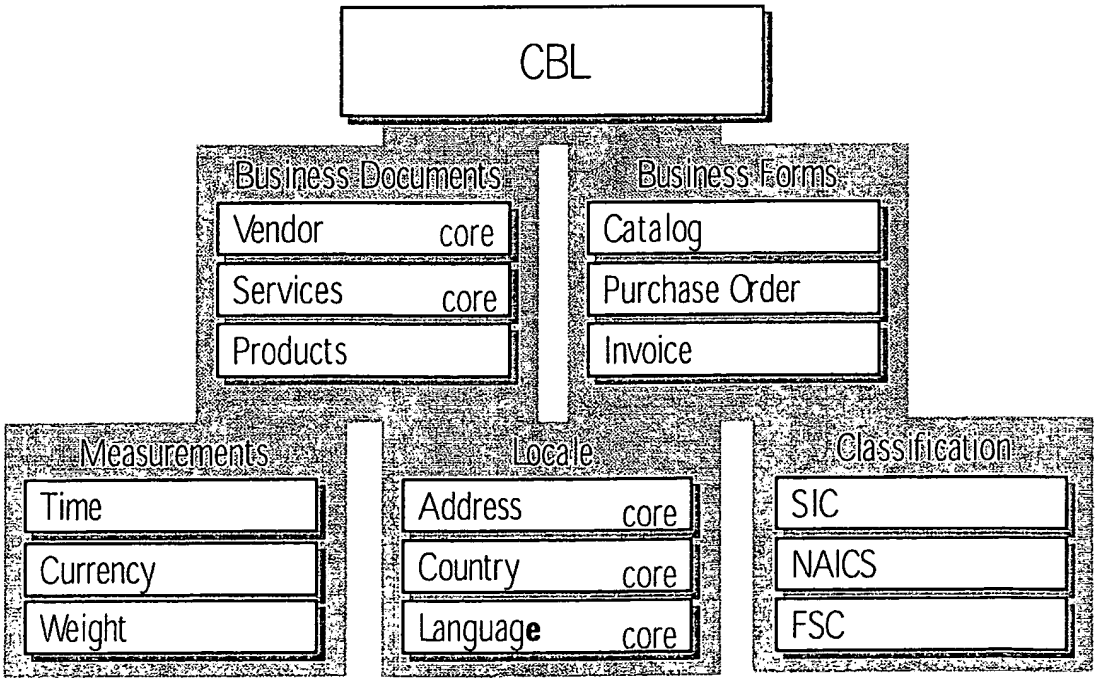
Open Framework For Commerce



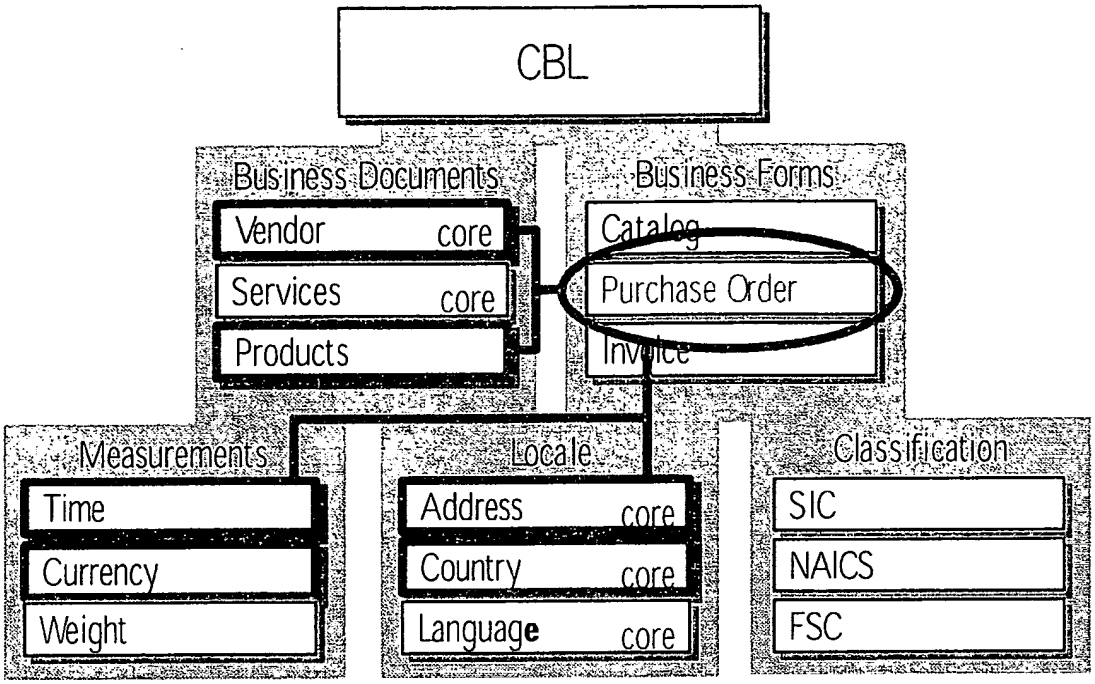
The Common Business Library

- The functions and information that are common to all business domains, building on existing standards or conventions
- Specifies common semantics, common syntax, and message packaging
- CBL documents are described by XML DTDs to make them "self-descriptive" and validatable
- Complex descriptions and messages can be composed from primitives
- Domain-specific XML applications can be implemented in "native" form or as "hybrids" for maximal interoperability

Building Blocks



Building Blocks



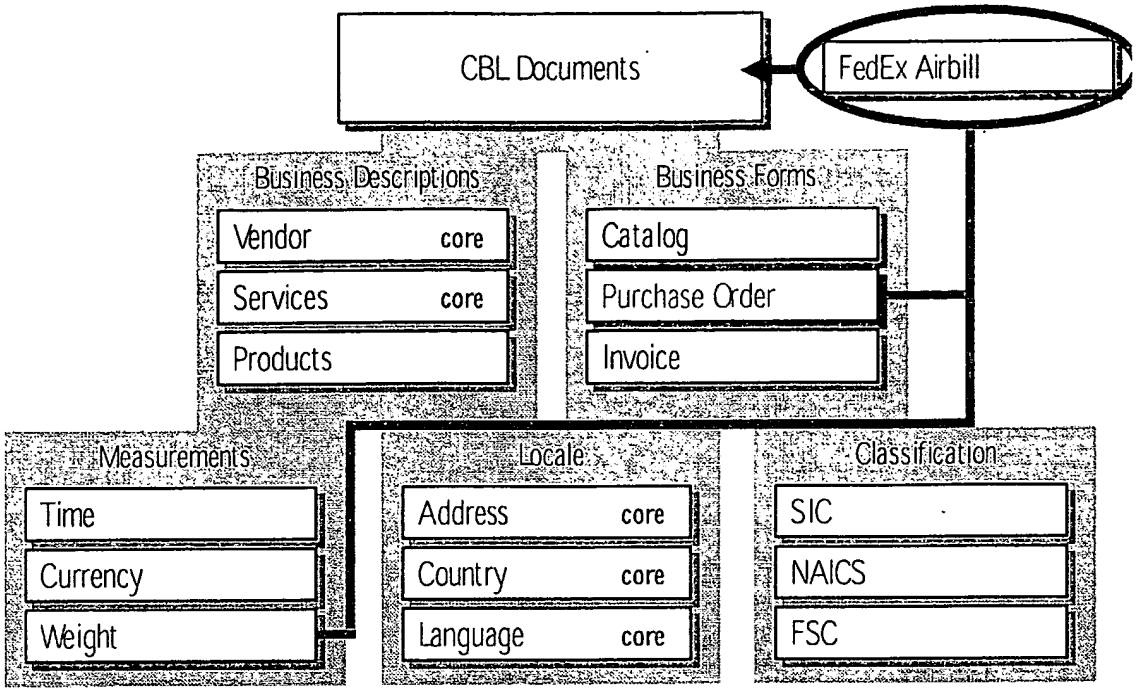
v|e|o 27

Document Type Definitions and Modules

- addresso.mod, for Geographical Information
- catentry.dtd, for a Simple Catalogue Entry
- contact.mod, for Contact Information
- countrys.mod, for Country Codes
- datetime.mod, for Date and Time markpart.dtd, for Market Participant Information
- markdesc.dtd, for a Market Description
- pay.mod, for Payment Information
- proddesc.dtd, for a Simple Product Description
- shipment.mod, for Shipping Information
- [illegible] for a Shopping Cart
- transact.dtd, for Transaction Documents

v|e|o 28

CBL Building Blocks



Business Services Described Using CBL

```
<service>
<service.name>Order Service</service.name>
<service.location>www.veosystems.com/order</service.location>
<service.op>
        <service.op.name>Submit Order</service.op.name>
        <service.op.inputdoc>po.dtd</service.op.inputdoc>
        <service.op.outputdoc>poack.dtd</service.op.outputdoc>
</service.op>
< service.op>
        < service.op.name>Track Order</service.op.name>
        <service.op.inputdoc>request.track.dtd<service.op.inputdoc>
        <service.op.outputdoc>response.track.dtd<service.op.outputdoc>
</service.op>
</service>
```

CBL Status

- CBL v1.0 contains a few dozen DTDs and modules developed from analysis of ISO, ANSI X.12, other standards
- CBL currently being used by Veo Systems in demonstration applications (Project Seitai, GSA catalog interoperability)
- CBL to be starting "fodder" for CommerceNet-sponsored WG to develop open framework for interoperability of domain-specific commerce languages (just getting under way)

Interoperable Documents

The Economist

"Untangling the Web"

25 April 1998

.."But the biggest role that XML is expected to play is in integrating the way that existing paper documents -- invoices, loan applications, contracts, insurance claims, you name it are exchanged between organizations around the world. Imagine what the world would be like if one company's computer system could automatically read any other organization's documents - and make complete sense of them? This is the goal that the technique known as EDI has struggled, unsuccessfully, to achieve for years. Though efforts have barely begun, there is a chance that XML could actually make that happen. If it did, business on the Web could run riot."

ATTACHMENT:

GLUSHKO 1999 ARTICLE

ROBERT J. GLUSHKO, JAY M. TENENBAUM,
AND BART MELTZER

AN XML FRAMEWORK FOR Agent-based E-commerce

Emerging standards for commercial document exchange promise open business-to-business e-commerce.

COMMERCENET'S ECO SYSTEM INITIATIVE, LAUNCHED IN 1996, aims to transform the World-Wide Web into an agent-based infrastructure for Internet commerce.

Today's Web gives people unprecedented access to online information and services. But its information is delivered in format-oriented, handcrafted hypertext markup language (HTML), making it understandable only through human eyes. Software agents and search engines have difficulty using the information because it is not semantically encoded. Clever programmers work around some of HTML's inherent limitations by using proprietary tags or software that "scrapes" Web pages to extract content. Unfortunately, such ad hoc approaches do not scale. Proprietary tags require browser plug-ins, and scraping approaches require a customized script for each Web site. These approaches balkanize the Web, making it inaccessible to agents.

Tomorrow's Web will use the extensible markup language (XML) to encode information and services with meaningful structure and semantics that computers can readily understand. In Internet commerce, companies will use XML documents for publishing everything from product catalogs and airline schedules to stock reports and bank statements. They will also use XML forms to place orders, make reservations, and schedule shipments. Any agent with the proper authorization will be able to obtain computer-interpretable data sheets, price lists, and inventory reports through the Web or email, then request quotes, place orders, and track shipments.

By making the Web accessible to agents and other automated processes, XML will fundamentally transform the nature of e-commerce (see Maes et al.'s "Agents That Buy and Sell" in this issue). XML will eliminate the need for custom interfaces with every customer and supplier, allowing buyers to compare products across many vendors and catalog formats, and sellers to publish their catalog information once to reach many potential buyers. Online businesses will also be able to build on one another's published content and services to create innovative virtual companies, markets, and trading communities.

Web merchants might initially dread that XML-encoded information makes it too easy for buyers to compare prices and competitors to co-opt their content. But fear of lost business opportunity as e-commerce grows and the recognition that XML provides many other advantages for sellers (such as the ability to differentiate products in ways other than price) are likely to convince them to adopt richer markup formats. (see Wong et al.'s "Java-based Mobile Agents" in this issue). In time, most merchant Web sites will provide agent-searchable catalogs that supply product descriptions, as well as information about price and availability.

For consumers, the most obvious result of pervasive markup will be smart shopping agents that level the playing field in their dealings with sellers. Using Internet-wide shopping directories, these agents will be able to locate all merchants carrying a specific product or service, then query them in parallel to locate the best deals. Some merchants will provide sales agents that negotiate with shopping agents and generate customized offers in response to their solicitations. The shopping agents can then sort the offers they receive according to criteria set by their owners—the cheapest flight, the most convenient departure time, the roomiest aircraft, or some weighted combination. Cybermediaries will offer innovative brokering and referral services that match buying and selling agents, as well as order-aggregation services that increase their purchasing clout.

Agent-based shopping by consumers is just the tip of the e-commerce iceberg. Whenever a product is bought, information propagates back down the supply chain, triggering a series of distribution, manufacturing, and logistics events. Today much of this business-to-business information is exchanged through EDI messages. But traditional EDI is complex and expensive, because most messages travel over proprietary networks. Moreover, EDI's brittle syntax necessitates a custom integration solution between each pair of trading partners.

For these reasons, EDI transactions will increasingly take place over the Internet using an XML/EDI message format. Such messages will be more economical than traditional EDI messages, while being easier to validate and translate into the formats needed by applications at each end of the exchange [4]. This development will encourage businesses, including many that find traditional EDI too costly, to implement Web agents that respond to XML messages. This agent-based approach to enterprise integration is simpler and more open than traditional EDI, because it avoids the "pairwise tyranny" through which big companies impose proprietary message formats on small companies. More-

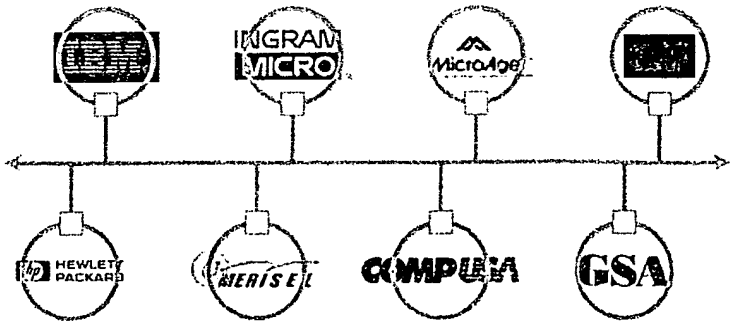


Figure 1. A supply Web linking PC manufacturers, distributors, and resellers

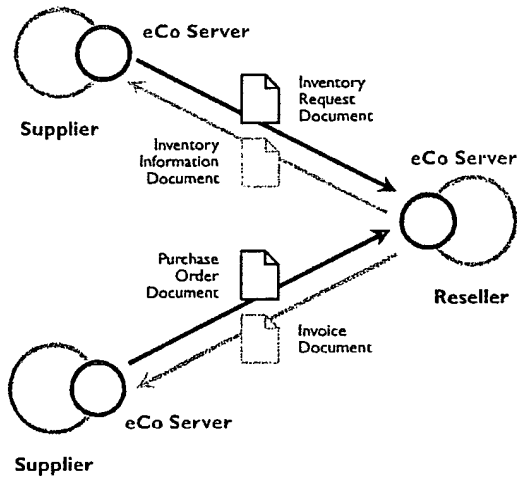


Figure 2. XML-based document exchange in the eCo System

over, publishing XML-encoded documents, such as data sheets and price lists, on the Web makes the information available instantly to all potential trading partners. Instant availability transforms rigid supply chains into "supply Webs," in which participants transact business spontaneously (see Figure 1).

The eCo System began as an architectural vision for open Internet commerce [5], proposed and evangelized by the 500-member worldwide CommerceNet Consortium in 1996. Conceived originally as a CORBA-based interoperability framework, the eCo System architecture was recast in 1997 on an XML foundation, due to XML's simplicity and widespread adoption by key vendors, including IBM, Microsoft, Netscape, and Sun.

Today's eCo System enables companies to communicate over the Internet using self-defining XML business documents that agents, as well as people, can easily understand. Business Interface Definitions (BIDs), posted on the Web, tell potential trading partners what online services a company offers and what documents to use when invoking those services. For example, a BID might allow a customer to order goods by submitting a purchase order or a supplier to check availability by downloading an inventory status report (see Figure 2).

A key element of the eCo System framework is the Common Business Library (CBL), an extensible, public collection of generic BIDs and document templates that companies can customize and assemble to go online quickly.[1] CBL includes XML message templates for the basic business forms used in ANSI X12 EDI transactions, as well as those used in such emerging Internet specifications as Open Trading Protocol (OTP) and Open Buying on the Internet (OBI). These specifications are mapped to each other using a dictionary of common business terms and data elements. A company can thus define its business interface in terms of any Internet standard mapped to CBL and communicate instantly with every other company that has done the same, even when the companies subscribe to different standards.

The eCo System framework overcomes two longstanding barriers to e-commerce. CBL facilitates spontaneous commerce between trading partners without custom integration or prior agreement on specific industrywide standards. And by being interpretable by both people and agents, XML documents provide an incremental path to business automation, whereby browser-based tasks are gradually transferred to computer agents. These advances eliminate much of the time, costs, and risks of traditional system integration. Moreover, the eCo System transforms closed trading partner networks into open markets and extends such enterprise applications as inventory management and production scheduling across entire supply chains.

XML is a simplified metalanguage, derived from SGML, emerging as the standard for self-describing data exchange in Internet applications. XML was developed by the World-Wide Web Consortium in 1997 and is being implemented rapidly by such major platform vendors as IBM, Microsoft, Netscape, and Sun Microsystems. XML's power

[1]The CBL was called the Common Business Language in earlier descriptions of eCo System. The change emphasizes CBL's function as a set of building blocks for XML applications and its role as a complement (rather than as a competitor) to ICE, OBI, OFX, OTP, RosettaNet, and other commerce languages.

derives from its extensibility and ubiquity. Anyone can invent new tags for particular subject areas, defining what they mean in document type definitions (DTDs). Content-oriented tagging enables a computer to understand the meaning of data, including, say, whether a number represents a price, a date, or a quantity.

This tagging significantly increases the functionality of Web e-commerce applications, because they can now do much more than simply display product data. For example, items in an XML-encoded catalog can be sorted by price, availability, and size.

One of eCo System's longstanding goals has been to enable businesses to build on one another's services to create virtual enterprises. Such plug-and-play commerce involves modeling enterprises as collections of services, some internal to a particular business, others provided by trading partners. Business services in eCo were originally defined as CORBA application programming interfaces (APIs). While the CORBA approach appears workable within organizations that control APIs, our experience in several prototypes suggests it is not practical for interenterprise integration. Fortunately, XML offers a promising alternative—agents interacting with business services through business documents.

Business documents represent a more intuitive and flexible way to access business services than programming APIs. It is much easier to interconnect companies in terms of the documents they exchange, on which they already largely agree, than in terms of their business system interfaces, which invariably differ. The coupling is looser, but loose coupling is better than no coupling at all.

XML's human readability is another significant advantage over CORBA. Just as HTML is a language for the eyes, CORBA is a language for CPUs, meant to convey information among programs, with no concession to human readability. XML documents are as readily interpretable by humans as they are by computers, especially with the aid of a style sheet [2].

Other proposals for agent languages suggest that first-order logic or other formal languages enable more precise specification of messages than XML [1, 3]. We prefer XML for two reasons—one language-theoretic, one practical. Expressing semantics in syntax rather than in first-order logic leads to a simpler evaluation function while needing no agreement on the associated ontologies. The practical argument, which is much more important for commercial success, is XML's ubiquity. The Web has made everyone appreciate the power of markup languages, practically assuring the widespread adoption of XML, as

Domain-specific Commerce Languages

The power of XML in enabling interoperability and simplifying the sharing and reuse of information between business domains is encouraging companies to work together to develop XML-based specifications for the business information they exchange most often. Sample specifications include:

- Open Trading Protocol. A consortium of banking, payment, and technology companies is specifying information requirements for payment, receipts, delivery, and customer support (www.otp.org). The goal of OTP is efficient exchange of information when the merchant, the payment handler, the deliverer of goods or services, and the provider of customer support are different entities with their own systems.
- XML/EDI. A group chartered jointly by CommerceNet, ANSI X12, and the Graphics Communication Association is defining how traditional X12 EDI business data elements should be represented using XML (www.xmledi.com).
- RosettaNet. This PC industry initiative is defining how to exchange PC product catalogs and transactions among manufacturers, distributors, and resellers (www.rosettanet.org).
- Open Buying on the Internet. The OBI initiative, launched by American Express and major buying and selling organizations, including Ford Motor and Office Depot, is automating large-scale corporate procurement of office and maintenance supplies (www.openbuy.org)
- Information and Content Exchange. CNET, News Corp., Vignette, and other information content providers are developing ways through ICE to create and manage networked relationships, such as syndicated publishing networks, Web superstores, and online reseller channels (www.w3.org/TR/1998/NOTE-ice-19981026).
- Open Financial Exchange. Originally proposed by CheckFree, Intuit, and Microsoft for the electronic exchange of financial statements among consumers, small businesses, and financial institutions, the OFX effort supports banking, bill payment, investment, and financial planning activities (www.ofx.net).

Share the Ontology in XML-based Trading Architectures

First bring semantic order to the world of XML.

Howard Smith and Kevin Poulter

Recent e-commerce application activity involving the extensible markup language (XML) has led to a proliferation of XML-based standards and markup language proposals. Among them are several designed to support site-to-site Web automation that lean naturally toward the agent paradigm of distributed computation.

Although XML represents a major step forward in e-commerce technology, business-to-business trading partners should also recognize XML's limitations. XML is not a cure-all for system interoperability, but a widely accepted foundation layer on which to build. Moreover, there are differing views on how to extend or complement XML to support agent-based e-commerce (see Glushko et al.'s "An XML Framework for Agent-based E-commerce" in this issue). This challenge is further complicated by debate over some fundamental questions: How should XML be extended to support the representation of business information? Should XML be enriched with tags reflecting higher-level concepts, especially business domains, such as standard business processes? How should foundation ontologies (from which higher-level content is composed) be defined? How can the numerous heterogeneous e-commerce frameworks (such as ICE, OBI, OTP, and XML/EDI) be unified to enable the expected low-friction market of the future? And will the future electronic marketplace be dominated by a series of commerce islands with trading groups isolated by the proprietary protocols and domain models with which their commerce agents interact?

Answers involve not only solving the related technology and intellectual challenges, but how to bring together the various communities of industrial standards developers. Each holds the essential elements of the overall solution. These communities, including EDI, Internet, knowledge engineering, and SGML, bring to the table subtly differing angles on the problem, including representation approaches associated with rich documents, publish/subscribe protocols, transactions, content syndication, and business semantics. To survive in this market, e-commerce component providers will have to support a number of different content formats and transaction frameworks, translating among them to achieve significant penetration. It appears that the main barrier to e-commerce lies in the need for applications to share information, not in the Internet's reliability and security.

Due to the wide range of enterprise and e-commerce systems being deployed by businesses and the way these systems are variously configured, the problem is particularly acute among large electronic trading groups. E-commerce will increasingly focus on trans-enterprise communication, while the number of trading partners and sophistication of e-commerce applications also increase. The need to unite business models, processes, and representation formats is greater than ever, while expectations run ever higher. Although many companies have already begun to organize, standardize, and stabilize their digital services in order to create and maintain sustainable network relationships with their trading partners, they are doing so only in conjunction with their immediate trading partners. This relatively narrow focus can limit the return on investment possible from each of these initiatives.

A global environment. There is now a need for e-commerce participants to create a global environment providing significant interoperability between the systems used by all engaged. Such an environment can be achieved through improved semantics within Internet transactions and in networked service definitions. It will facilitate consistent behavior among participants in large trading networks or within complex virtual organizations. Many of the foundation concepts needed to achieve this consistent behavior have already been established through work on distributed problem solving, intelligent agents, and knowledge sharing, yet to date these technologies have had little effect on Internet-based commerce.

Agent-based systems to support the next generation of Internet commerce must adopt common ontologies if they are to interact without misunderstanding. For example, content can be defined to enable application interoperation as well as information synthesis. An e-commerce standard being developed by major PC vendors, resellers, and distributors has shown by practical example in the PC distribution chain that quite sophisticated representation issues can complicate even straightforward commerce scenarios. For example, the required catalog model includes the need to represent the topology of the parts comprising a PC product.

But to bring semantic order to the world of XML, we have to be clear about what we mean by "ontology." The term is often used to refer to a vocabulary, yet even the terms within a simple vocabulary can be prone to misinterpretation, particularly in combination, unless they have been chosen carefully. Consider some of the problems already apparent in the plethora of e-commerce standards that

have emerged during the past few years. As new online trading environments are developed, the potential protocol mismatches between participants' commerce platforms can become major inhibitors to achieving industrywide e-commerce solutions and delivering supply-chain and market-efficiency benefits. Realizing Web automation in such complex environments reopens many of the problems and issues the knowledge-sharing and intelligent-agent communities have been wrestling with in such initiatives as the shared design environment, or SHADE, and the advanced technology operations system, or ATOS, using ontologies to enable agents working on different problems to interoperate over networks.

XML as a representation is just too forgiving at the document type definition (DTD) stage at the expense of the information processing stage. However, steps are being taken in the right direction; an example is the definition of schema languages to enable consistent schema semantics in the definition of objects in XML (such as by the World-Wide Web Consortium reflecting proposals from a number of organizations).

Consistent schema semantics will certainly enable efficient e-commerce using predefined DTDs between fixed networks of trading partners. But to enable the full benefits of agent-based e-commerce—where agents act in an autonomous or semiautonomous way, comparing and contrasting products or suppliers and negotiating with other agents—participating agents have to communicate in terms of a detailed ontology of the business domain.

The challenge for technology vendors, e-commerce participants, and standards bodies is to capitalize on the experience available in the knowledge representation and distributed agent communities.

Veo Systems is pursuing a pragmatic approach to solving some of these issues through the Common Business Library, an extensible, public collection of business interface definitions and document templates. This library is being rationalized and further developed by the CommerceNet eCo Framework Working Group established last year and should provide a foundation for addressing many of the unanswered questions in agent-based e-commerce. Ontologies will play a key role.

HOWARD SMITH (howard.smith@ontology.org) is the director of Ontology.Org and a principal consultant (Internet) in Computer Sciences Corp. in Farnborough, Hampshire, U.K.
KEVIN POULTER (kevin.poulter@ontology.org) is chief technology officer of Ontology.Org and a principal consultant in Computer Sciences Corp., U.K.

HTML's heir apparent. XML may be theoretically less expressive than other formal languages, but we prefer a language that can be understood and produced by computer novices to a theoretically better one spoken only by computer scientists.

The significance of XML for integration extends beyond the Web to email, database records, and programming APIs. An XML parser imposes the same API on any XML data source, eliminating much of the need for custom programs to extract and integrate information from each source. So, integrating enterprise information from accounting, purchasing, manufacturing, shipping, and other functions can be accomplished by first converting each source to XML and then processing the parsed data stream. Put another way, each application need know only two source formats—its own and XML—rather than having to produce the native format of every other application.

XML by itself doesn't enable plug-and-play commerce. In addition to the language itself, a complete business integration solution also requires: standardized tags, or metadata, for each commerce community; a means for mapping between different metadata descriptions; and a server for processing XML documents and invoking appropriate applications and services. The eCo System framework starts with XML and adds these additional architectural and technology elements.

Specialized Markup Languages

XML makes it easy to create specialized markup languages that identify and describe buyers and sellers, the goods and services they want to buy or sell, and the various other document types involved in commerce. However, a vendor has obvious incentives for describing its offerings in ways that highlight its competitive advantages and that obscure comparison on features where it lacks an advantage. But if every business invented its own XML definitions for product catalogs, requests for quotes, price lists, purchase orders, invoices, transportation schedules, shipping notices, and delivery and payment receipts, the Web would be scarcely more usable as a platform for agents and other automated processes than it is today (see Smith's and Poulter's "The Role of Shared Ontology in XML-based Trading Architectures" in this issue).

Fortunately, many companies already recognize the need for information-exchange standards, uniting in several initiatives focusing on XML standards for particular industries or business

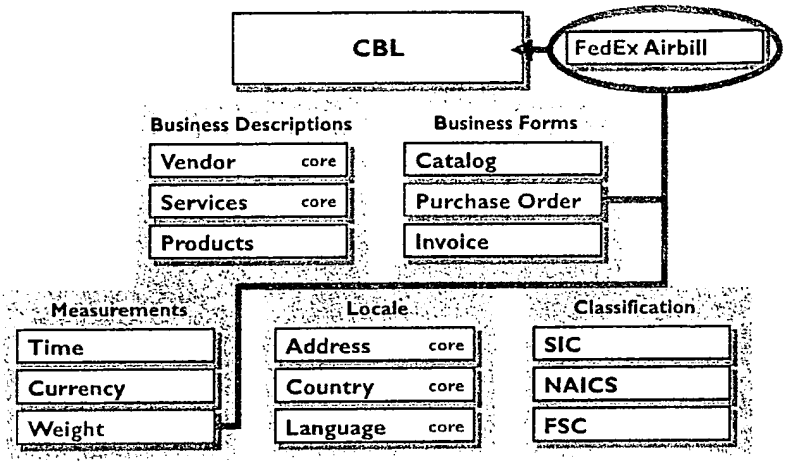


Figure 3. The Common Business Library

processes (see the sidebar "Domain-specific E-commerce Languages"). Unfortunately, these initiatives operate independently, doing little to facilitate interaction across industry and functional boundaries. The solution is to spur development of XML document models based on reusable semantic components common to many business domains. Such documents can be understood by any business through their common elements (such as address, date, and part number), while also providing a common mechanism for linking to the unique elements vendors need to differentiate themselves.

The CBL is designed to encourage development and use of generic XML document models. The library consists of information models for various concepts, including:

- Business descriptions, such as companies, services, and products;
- Business forms, such as catalogs, purchase orders, and invoices; and
- Standard measurements, such as date and time, location, and classification codes.

These models are represented as an extensible, public set of XML building blocks that companies can customize and assemble to develop XML applications quickly. Atomic CBL elements implement industry messaging standards and conventions, such as standard International Organization for Standardization (ISO) codes for countries, currencies, addresses, and time. Low-level CBL semantics are also derived through analysis of proposed metadata frameworks for Internet resources, such as the Dublin Core metadata element set developed by the Online Computer Library Center.

The next level of CBL elements use these building blocks to implement the basic business forms used in X12 EDI transactions, as well as those in OTP, OBI, and other emerging Internet standards.

A working group organized by CommerceNet and

```
<service>
<service.name>Order Service</service.name>
<service.location>www.veosystems.com/order</service.location>
<service.op>
<service.op.name>Submit Order</service.op.name>
<service.op.inputdoc>www.commerce.net/po.dtd</service.op.inputdoc>
<service.op.output>www.veosystems.com/invoice.dtd</service.op.outputdoc>
</service.op>
<service.op>
<service.op.name>Track Order</service.op.name>
<service.op.inputdoc>www.commerce.net/request.track.dtd<service.op.inputdoc>
<service.op.outputdoc>www.veosystems.com/response.track.dtd<service.op.outputdoc>
</service.op>
</service>
```

Figure 4. Fragment of an XML service definition for an eCo-compliant business application

other organizations recently began using CBL to create a base set of common terms, or mappings, between existing terms in commerce specifications, including OBI and OTP. The final result scheduled for release in mid-1999 will include a recommended base set of XML data elements, attributes, and definitions for use in e-commerce standards initiatives; they will be made freely available in public registries run by CommerceNet and other organizations. The Internet community, building on this foundation, will be encouraged to contribute additional elements and document models.

Agent-based shopping by consumers online is just the tip of the e-commerce iceberg.

Figure 3 shows how Federal Express might use CBL to create an XML version of its airbill by customizing a generic purchase order DTD with specific information about shipping weight. The generic purchase order, in turn, is assembled from more primitive CBL modules for address, date and time, currency, and vendor and product description. This example shows how reusing CBL components can significantly speed development of XML e-commerce applications and facilitate their interoperation.

When creating CBL, we found it helpful to extend XML with a schema language. The extensions add strong typing to XML elements so content can be readily validated. For example, an element called `CPU_clock_speed` can be defined as an integer with a set of valid values: {100, 133, 166, 200, 233, 266 Mhz}. The schema language also adds class-subclass hierarchies, so information is readily instantiated from class definitions. A laptop, for instance, can be described as a computer with additional tags for such features as display type and battery life. These and other extensions facilitate data entry, as well as automated translations between XML and traditional object-oriented and relational data models.

Trading partners not only have to agree on the meaning of message tags but understand how to use them for conducting business. In the eCo System, BIDs tell potential trading partners what online business services a company offers and which documents to use when invoking those services. In effect, services are defined by the documents they accept and produce. BIDs present a clean and stable interface to business partners, insulating them from a company's internal changes in technology, organization, and processes.

Figure 4 shows a fragment of a BID, defining an XML service for an eCo-compliant business. The service definition consists of two transactions—one for taking orders, one for tracking them. Each definition expresses a contract, or promise, to carry out a service if a valid request is submitted to the specified Web address. The order service requires an input document conforming to a standard `po.dtd` DTD in an industry registry operated by CommerceNet. If the service is able to fulfill the order, it returns a document conforming to a customized `invoice.dtd` whose definition is local. In effect, the company is promising to do business with anyone submitting a purchase order conforming to the XML specification it declares. No prior arrangement is needed.

A DTD is the formal specification, or grammar, for documents of a given type, describing the elements, their attributes, and the order in which they have to appear. For example, purchase orders typically include the names and addresses of the buyer and seller, a set of product descriptions, and associated terms and conditions, such as price and delivery dates. In the EDI world, the X12 850 specification is a commonly used model for purchase orders.

From Business Services to Virtual Enterprises

eCo servers provide the glue that links a set of internal and external business services to create a virtual enterprise or trading community. The server parses incoming documents and invokes the appropriate services (as specified by the applicable BID) by, say, handing off a request for product data to a catalog server or forwarding a purchase order to an enterprise resource planning system. The eCo server also handles translation tasks, mapping the information from one company's XML documents onto document formats used by its trading partners and into data formats required by its own legacy systems.

Following the service definition in Figure 4, when a company submits a purchase order, the XML parser in the eCo server uses the purchase order DTD `po.dtd` to transform the purchase order instance into a stream of information events. These events are then routed to any applications programmed to handle events of that type; in some

cases, the information is forwarded over the Internet to an entirely different business. In the purchase order example, information coming from the parser may be acted on by various applications:

- An order entry system processing the purchase order as a complete message;
- An enterprise resource planning system checking inventory for the products described in the purchase order;
- A customer database verifying or updating a customer's address;
- A shipping company system using the address information to schedule a delivery; and
- A bank system using credit card information to authorize a transaction.

However, what is most important in such processing is what is left out. Trading partners need agree only on the structure, content, and sequencing of the business documents they exchange, not on API details. How a document is processed and what actions result are strictly up to the business providing the service. This focus on commerce elevates enterprise integration from the system level to the business level.

A True Marketplace

eCo System's top-level goal is to transform the Web into a true marketplace by enabling spontaneous, peer-to-peer exchange of electronic business documents among all companies. This document-based approach replaces complex, expensive, and proprietary business integration solutions with one that is simple, affordable, and open.

The eCo architecture recognizes that a single dominant e-commerce standard is unlikely, even within a particular business community (and certainly not across communities). Rather, there will be many standards. CBL, in particular, is not a single standard but a collection of common business elements underlying all EDI and Internet commerce protocols. Its reusable components speed implementation of standards and facilitate interoperation by providing a common semantic framework. This approach to standards implementation and interoperation is fundamentally different from that taken historically by standards organizations and software vendors. It occupies an openness high ground embracing all the new competing standards being developed to take advantage of XML.

The eCo system framework and CBL are being evaluated in several of the standards initiatives listed in the sidebar on domain-specific commerce languages, as well as two major market trials sanctioned by CommerceNet:

- The U.S. General Services Agency (GSA). The largest buying organization in the U.S., GSA is creating catalog interoperability across numerous government agencies. Until now, the catalogs belonging to participating agencies were implemented as relational databases, as static files, or as catalog applications. An eCo server transforms each of these information sources into a standard catalog service that responds to CBL queries by outputting an XML data stream conforming to a common catalog schema. The integrated source catalogs can then be searched through specialized user interfaces developed by various participating technology vendors.
- RosettaNet. The RosettaNet consortium of PC manufacturers, resellers, and distributors is developing integration standards for the PC distribution channel; participants include Compaq Computer, CompUSA, Dell Computer, Hewlett-Packard, IBM, Ingram Micro, Merisel, Microsoft, and Tech Data.

The XML document models used in these initiatives are being rationalized to identify common semantic elements. These elements will be added to various public CBL repositories and made freely available (for more detail, visit www.commerce.net and www.veosystems.com). ■

REFERENCES

1. Finin, T., Fritzson, R., McKay, D., and McEntire, R. KQML as an agent communication language. CIKM '94. In *Proceedings of the Third International Conference on Information and Knowledge Management*, 1994, pp. 456–463.
2. Fuchs, M. Domain-specific languages for ad hoc distributed applications. In *Proceedings of the Conference on Domain-Specific Languages*, 1997.
3. Kimbrough, S., and Moore, S. On automated message processing in electronic commerce and work support systems: Speech act theory and expressive felicity. *ACM Trans. Inf. Syst. 15*, 4 (Oct. 1997), 321–367.
4. Laplante, M. Making EDI accessible with XML. *EC.COM 4*, 2 (March 1998), 23–26.
5. Tenenbaum, J., Chowdhry, T., and Hughes, K. eCo System: An Internet commerce architecture. *Comput. 30*, 5 (May 1997), 48–55.

ROBERT J. GLUSHKO (glushko@veosystems.com) is director of information engineering at Veo Systems, Inc., in Mountain View, Calif.
JAY M. TENENBAUM (jmt@veosystems.com) is chairman and chief scientist of Veo Systems, Inc., in Mountain View, Calif.
BART MELTZER (bart@veosystems.com) is chief technology officer of Veo Systems, Inc., in Mountain View, Calif.